Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

Unsupervised Learning of HTNs in Complex Adversarial Domains

Michael Leece
University of California, Santa Cruz
mleece [at] ucsc.edu

Abstract

While Hierarchical Task Networks are frequently cited as flexible and powerful planning models, they are often ignored due to the intensive labor cost for experts/programmers, due to the need to create and refine the model by hand. While recent work has begun to address this issue by working towards learning aspects of an HTN model from demonstration, or even the whole framework, the focus so far has been on simple domains, which lack many of the challenges faced in the real world such as imperfect information and real-time environments. I plan to extend this work using the domain of real-time strategy (RTS) games, which have gained recent popularity as a challenging and complex domain for AI research.

Introduction

HTNs are a very popular planning framework, and have been used in many real-world domains, from manufacturing to medical care. However, there is one great drawback traditionally cited to them whenever they are considered, and this is the cost of expert labor involved in creating a useful HTN. Some work has been done in the learning of HTN method pre/postconditions, and even of learning the full method decomposition structure from scratch. However, these have focused on extremely simplistic domains such as the classical Blocks World or logistics transportation proglem.

In light of this, I plan to extend the work on HTN learning into more complex domains. In particular, I am working with RTS games, a popular video game genre in which players compete in an economic and military simulation to defeat their opponents. This domain has a number of properties that increase its complexity over standard planning domains; namely, they are real-time, adversarial, and imperfect information. In addition, the state/action space is many orders of magnitude greater than prior domains.

In addition to working in a more complex domain, my goal is to learn from human demonstrations, rather than planner-generated traces. This introduces new challenges in the form of unpredictability and lack of knowledge of the agent's internal model, but is necessary when working in domains where the hand-authoring of a planner is prohibitive.

Completed Work

Thus far, I have worked on two approaches to the problem of learning HTNs from human demonstrations in the RTS domain. The first uses Generalized Sequential Pattern mining (GSP) to search for common patterns of actions from a database of human replays, working off of the assumption that common sequences of actions that are closely linked in the time domain present promising HTN method abstractions. These patterns are then replaced with an abstract symbol representing the method, and the process is repeated. This results in a hierarchy of methods that decompose into primitive actions. They lack preconditions, but in theory those could be learned given the method structure. The initial work on this approach was presented in (Leece and Jhala 2014), though it has progressed since then.

The second approach is an expectation maximization algorithm. At a high level, it proposes a small number of goals a player could be pursuing at any given time, modeled as Markov transition functions, initialized randomly. Given a database of traces, it then assigns actions to goals in order to maximize the likelihood of each trace occurring. From there it follows the standard EM iteration, redefining the goals' transition functions based on their assigned actions, and repeating. At the end, the abstract goal that has the least entropy (representing the least aggregated noise) is added to the database, and others are discarded. Actions assigned to the added goal are then replaced with its abstract symbol, and we iterate on the updated database.

In general, one of the challenges that I have faced is a way to rigorously evaluate the results of these algorithms. I have recently implemented an agent that uses HTN planning to play RTS games, but there are many confounding factors when using pure strength of gameplay as a metric. I would welcome advice on this topic from any mentor.

A final piece of completed work is some research on learning meta-reasoning strategies from human demonstrations in RTS games that I am currently working on as a collaboration with the Naval Research Laboratory. As a work in progress, I cannot go into technical detail, but we hope to publish it in ICAPS '16.

Proposed Work

Both of the approaches described above still have a number of challenges to address. For the GSP approach, a method must be determined to identify when two slightly different patterns are actually achieving the same abstract goal. Much of the flexibility from HTN planning comes from having multiple methods to achieve the same goal, and the lack of this in the generated models is an issue. In addition, the problem propogates upward through the tree, and patterns that should be found are not, if two methods are not identified as achieving the same goal. Additionally, I need to learn preconditions for the methods. While this has been done before, it was in less noisy environments with consistent actors, which may complicate the process.

For the EM approach, the learned abstract goals do not translate directly to HTN methods, since they are modeled as Markov transition functions. Either they must be translated into HTN methods, using most probable paths as method sequences, or the HTNs themselves must be modified to adapt to a new type of method.

Finally, I would like to be able to make some formal guarantees regarding the models generated by these approaches. While it can be difficult to guarantee things when using human input as training, I believe it should be possible to present a set of reasonable assumptions regarding the input data, under which guarantees about the resulting models learned can be made. However, I am not entirely sure how to go about this, and need more background research on the topic.

Related Work

In both (Garland, Ryall, and Rich 2001) and (Garland and Lesh 2003), the authors discuss the challenges and possible extensions for learning an HTN model based on annotated replays, and provide an algorithm that returns a provably sound and complete model that is consistent with the data (if possible).In a similar vein, (Ilghami et al. 2002) and (Ilghami et al. 2005) present work on learning HTN method preconditions, given the task structure decomposition, and succeed in some simple domains.

More recent work has moved toward the issue of attempting to learn the entire HTN model from traces, including the task structures. HTN-MAKER, proposed in (Hogg, Munoz-Avila, and Kuter 2008), presents an approach to deducing methods based on goal annotations that indicate when a goal has been achieved, and learning methods based on state/action matchings that move from a given state into the desired state. (Zhuo et al. 2009) and (Hogg, Kuter, and Munoz-Avila 2010) both present extensions of this work, respectively learning an action model/preconditions based on partial state observability, and integrating reinforcement learning into the HTN-MAKER framework.

Additionally, a number of systems with similar approaches have been developed for RTS games already. (Weber, Mateas, and Jhala 2012), (Weber 2012), (Ontañón et al. 2010) and (Ontanón et al. 2009) discuss two planning systems that use case based reasoning and a library of training data from humans to play RTS games, while (Ontañón and Buro 2015) presents an adversarial HTN approach. Outside of the RTS genre, (Hoang, Lee-Urban, and Muñoz-Avila 2005) developed a system for playing a first-person shooter game that used an HTN for strategic and tactical decision making, with a reactive system integrated to provide actual in-fight control.

Conclusion

In summary, my goal is to extend prior work on HTN learning into more complex domains that more accurately mirror the challenges that the real world has to offer. In addition, I am working on learning from human demonstrations rather than planner-generated traces, with the additional challenges that that brings.

References

Garland, A., and Lesh, N. 2003. Learning hierarchical task models by demonstration. *Mitsubishi Electric Research Laboratory (MERL), USA–(January 2002)*.

Garland, A.; Ryall, K.; and Rich, C. 2001. Learning hierarchical task models by defining and refining examples. In *Proceedings of the 1st international conference on Knowledge capture*, 44–51. ACM.

Hoang, H.; Lee-Urban, S.; and Muñoz-Avila, H. 2005. Hierarchical plan representations for encoding strategic game ai. In *AIIDE*, 63–68.

Hogg, C.; Kuter, U.; and Munoz-Avila, H. 2010. Learning methods to generate good plans: Integrating htn learning and reinforcement learning. In *AAAI*.

Hogg, C.; Munoz-Avila, H.; and Kuter, U. 2008. Htn-maker: Learning htns with minimal additional knowledge engineering required. In *AAAI*, 950–956.

Ilghami, O.; Nau, D. S.; Munoz-Avila, H.; and Aha, D. W. 2002. Camel: Learning method preconditions for htn planning. In *AIPS*, 131–142.

Ilghami, O.; Munoz-Avila, H.; Nau, D. S.; and Aha, D. W. 2005. Learning approximate preconditions for methods in hierarchical plans. In *Proceedings of the 22nd international conference on Machine learning*, 337–344. ACM.

Leece, M., and Jhala, A. 2014. Sequential pattern mining in starcraft: Brood war for short and long-term goals. 8.

Ontañón, S., and Buro, M. 2015. Adversarial hierarchical-task network planning for complex real-time games. In *IJCAI*.

Ontanón, S.; Bonnette, K.; Mahindrakar, P.; Gómez-Martín, M. A.; Long, K.; Radhakrishnan, J.; Shah, R.; and Ram, A. 2009. Learning from human demonstrations for real-time case-based planning.

Ontañón, S.; Mishra, K.; Sugandh, N.; and Ram, A. 2010. On-line case-based planning. *Computational Intelligence* 26(1):84–119.

Weber, B. G.; Mateas, M.; and Jhala, A. 2012. Learning from demonstration for goal-driven autonomy. In *AAAI*.

Weber, B. 2012. *Integrating learning in a multi-scale agent*. Ph.D. Dissertation, UC Santa Cruz.

Zhuo, H. H.; Hu, D. H.; Hogg, C.; Yang, Q.; and Munoz-Avila, H. 2009. Learning htn method preconditions and action models from partial observations. In *IJCAI*, 1804–1810.